

ABSTRACT

A new persistence format for storing objects of a user defined type in a database store enables information about the structure of the type to be communicated to the store. This information enables a number of store optimizations, including direct structural access to members of the type. Specifically, metadata is exchanged between the type implementer and the data store. The store uses the metadata to determine the storage layout for instances of the type. With this information, the store is able to detect access patterns that can be optimized to directly operate over the storage representation without hydration (deserialization) the object.